

## **REMARKS**

This communication is responsive to the Office Action mailed November 30, 2004. Claim 2 has been cancelled. Claims 1,5, 6-8, 11-20, 22 and 23 have been amended.

### **Specification**

A substitute specification is provided. In the substitute specification, the heretofore missing application serial number in paragraph [0001] is provided. In addition, the application has been amended with regard to capitalizing the JAVA trademark.

### **Claim Rejections, 35 USC § 101**

Claims 11-19 have been amended, as appropriate, to recite a “computer-implemented method.” It is respectfully submitted that these claims recite statutory subject matter.

### **Claim Rejections, 35 USC § 112**

The Examiner indicates that “Java runtime optimization” is not clearly defined. Applicant has amended this phrase, as appropriate, to “Java runtime environment optimization information” as suggested by the Examiner. This amendment is for clarification only, and does not affect the claim scope. The Examiner’s statement of the prior art rejection employs the language “Java runtime optimization information.” As discussed later with respect to the prior art rejections, Applicant has also included the word “environment” to clarify to what the “optimization” pertains.

With regard to “the last attribute” in claim 2, this claim has been cancelled.

With regard to the recitation of “first, last, and next optional attributes” in claim 5, it is believed that, particularly with reference to the specification at page 10, lines 21-24, it is clear that this phrase refers to functions included in the API for accessing the optional attributes. This language is not meant to recite that “each of the date items in the table is visited” as suggested by the Examiner.

With regard to the use of JAVA in the claims, it is respectfully submitted that this term has been used in a way that reasonably apprises one of ordinary skill in the art of the scope of the claims. The use of the term Java with respect to, for example, a programming language or virtual machine environment, indicates that the programming language or virtual machine complies with one or more well-known standards. This is clearly not the same as saying the JAVA trademark is descriptive of a product. To the extent a programming language, virtual machine, or other “entity” does not substantially comply with these standards, than it is not properly a “Java” entity. Thus, if anything, the use of the JAVA trademark in the claims serves to more clearly define the boundary of the claims. It is thus respectfully submitted that the use of JAVA in the claims does not render these claims indefinite.

### **Claim Rejections, 35 USC § 103**

Claims 1-18 and 20-25 stand rejected as being unpatentable over Cyran in view of Johnson. It is respectfully submitted that the rejection is improper and should be withdrawn.

In particular, it is respectfully submitted that the Examiner has mischaracterized the Cyran disclosure. The Examiner contends that Cyran discloses generating one or more optional attributes based on Java runtime optimization. However, what Cyran discloses optimizing is the compiled code itself (i.e., the native code). See, for example, col. 6, lines 48-59 (**bold** added), which discusses:

The JIT compiler, which operates in the contemplated limited resource environment of the code interpretive runtime system and whose execution speed is always critical to an application's performance, **then uses this pre-computed optimization information to operate optimized native code**. Therefore, the method of separating compilation in accordance with the present invention allows the JIT compiler to incorporate costly optimizations into its code generation process without actually incurring any of the resource-related expense that would otherwise be required if the separation was not done. This further allows the JIT compiler to continue operating within the stricter confinements imposed by its limited resource operating environment.

Clearly, optimizing the native code is not the same as optimizing the runtime environment. Examples of optimizing of the runtime environment discussed in the specification refer to loading particular features that are going to be used by a particular application. See, e.g., page 9, lines 21-24.

While the term “Java runtime” in the original claim is clearly intended to refer to the “Java runtime environment,” particularly in view of the description in the specification, Applicant has amended this term to explicitly refer to the runtime environment. In summary, at least since optimizing the native code is not the same as optimizing the Java runtime environment, the Cyran disclosure is inappropriate to the Examiner’s rejection.

The Examiner does not rely on the secondary reference, Johnson, as disclosing this feature. Thus, the Examiner has failed to make a proper *prima facie* case of obviousness with respect to claim 1.

With regard to claims 11 and 20, given the relevant similarities between claim 11 and 20, and claim 1, Applicant incorporates herein the discussion made above with regard to claim 1. It is respectfully submitted that the Examiner has failed to make a proper *prima facie* case of obviousness with respect to claims 11 and 20.

With regard to claim 19, the Examiner relies on the rejection of claim 18 as a foundational matter. As discussed above, the rejection of claim 18 is improper. Thus, for at least this reason, the rejection of claim 19 is similarly improper.

## CONCLUSION

Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,  
BEYER WEAVER & THOMAS, LLP



Alan S. Hodes  
Reg. No. 38,185

P.O. Box 70250  
Oakland, CA 94612-0250  
(650) 961-8300